

[illegible]

THE CYBER 70 ALGOL 68 COMPILER IMPLEMENTS THE LANGUAGE AS DESCRIBED IN THE "REVISED REPORT ON THE ALGORITHMIC LANGUAGE ALGOL 68", WITH THE FOLLOWING EXCEPTIONS:

IF A MULTIPLE VALUE V HAS A DESCRIPTOR WITH A FLEXIBLE STATE, THEN, ANY NAME N REFERRING TO A COMPONENT C OF V, IS TREATED AS IF IT WERE REFERRING TO A COMPONENT C OF A MULTIPLE VALUE WITH A DESCRIPTOR WITH A FIXED STATE. THIS IMPLIES THAT THE CONCEPT OF "TRANSIENT NAME" (RR 2.1.3.4.H) IS IGNORED IN THE IMPLEMENTATION. NAMES REFERRING TO COMPONENTS OF MULTIPLE VALUES WITH A DESCRIPTOR WITH A FLEXIBLE STATE, MAY BE ASSIGNED OR (VALUE REFERRED TO MAY BE) ASCRIBED TO OTHER NAMES, OR MAY BE ROWED OR DISPLAYED. HOWEVER, CARE SHOULD BE TAKEN WHEN THE VALUE V OF WHICH C IS A COMPONENT IS CHANGED BY ASSIGNATION (RR 5.2.1.2.C), BECAUSE THE VALUE TO WHICH THE NAME N AFTER THE ASSIGNATION REFERS, IS UNDEFINED, ALTHOUGH

WITH A FLEXIBLE STATE, MAY BE ASSIGNED OR (VALUE REFERRED TO MAY BE) ASCRIBED TO OTHER NAMES, OR MAY BE ROWED OR DISPLAYED. HOWEVER, CARE SHOULD BE TAKEN WHEN THE VALUE V OF WHICH C IS A COMPONENT IS CHANGED BY ASSIGNATION (RR 5.2.1.2.C), BECAUSE THE VALUE TO WHICH THE NAME N AFTER THE ASSIGNATION REFERS, IS UNDEFINED, ALTHOUGH NOT NON-EXISTING.

E.G. AFTER:

```
'FLEX' [1:0] 'INT' II := (1,2,3,4,5); 'REF' 'INT' J := II [3];  
II [1] = (6,7,8);
```

THE YIELD OF 'INT' {J} IS UNDEFINED, BUT NOT NON-EXISTING, BECAUSE

```
'REF' 'INT' {J} := 10; DOES NOT RESULT IN AN ERROR AND NEITHER  
DOES 'INT' K := J;
```

FURTHER MORE UPON DECLARATION OF A FLEXIBLE MULTIPLE OF VALUES CONTAINING FIXED MULTIPLES, INNER MULTIPLES ARE MADE FLEXIBLE.

THIS OBTVIATES THE NEED FOR A GHOST ELEMENT

E.G. 'FLEX' [1:5] [1:10] 'INT' AA;

IS INTERPRETED AS 'FLEX' [1:5] 'FLEX' [1:10] 'INT' AA;  
AND THUS THE ASSIGNATION AA[3] := (1,2,3) IS LEGAL, BECAUSE:

## B. FORMAL MODES.

NO DIFFERENCE IS MADE BETWEEN FORMAL MODES AND VIRTUAL MODES. THIS MEANS THAT EVERYWHERE IN A PROGRAM, WHERE THE REPORT REQUIRES A FORMAL DECLARER, ALL CONSTRUCTS WITHIN THAT DECLARER, THAT MAY NOT APPEAR IN A VIRTUAL DECLARER (BOUNDS, 'FLEX', 'EITHER') WILL BE IGNORED BY THE COMPILER. A WARNING MESSAGE WILL BE ISSUED.

THE ABOLITION OF FORMAL MODES IMPLIES THAT A DECLARATION SUCH AS

```
'FLEX' { : } 'INT' II = (1,2,3);
```

IS EQUIVALENT TO

```
{ : } 'INT' II = (1,2,3);
```

WHICH IT ESSENTIALLY ALREADY WAS, BECAUSE FLEXIBLE CONSTANTS ARE BY NO MEANS MORE FLEXIBLE THAN OTHER CONSTANTS.

NOTE THAT 'FLEX' CAN STILL APPEAR IN ACTUAL DECLARERS, AND THAT THE DECLARATION OF VARIABLES REMAINS THE ONLY PROPER WAY TO DECLARE A FLEXIBLE MULTIPLE.

## C. UNITED MODES.

E.G. 'FLEX' [1:5] [1:10] 'INT' AA;

IS INTERPRETED AS 'FLEX' [1:5] 'FLEX' [1:10] 'INT' AA;  
AND THUS THE ASSIGNATION AA[3] := (1,2,3) IS LEGAL, BECAUSE:

#### B. FORMAL MODES.

NO DIFFERENCE IS MADE BETWEEN FORMAL MODES AND VIRTUAL MODES.  
THIS MEANS THAT EVERYWHERE IN A PROGRAM, WHERE THE REPORT REQUIRES A FORMAL  
DECLARER, ALL CONSTRUCTS WITHIN THAT DECLARER, THAT MAY NOT APPEAR IN A VIRTUAL  
DECLARER (BOUNDS, 'FLEX', 'EITHER') WILL BE IGNORED BY THE COMPILER.  
A WARNING MESSAGE WILL BE ISSUED.

THE ABOLITION OF FORMAL MODES IMPLIES THAT A DECLARATION SUCH AS

'FLEX' [:] 'INT' II = (1,2,3);

IS EQUIVALENT TO

[:] 'INT' II = (1,2,3);

WHICH IT ESSENTIALLY ALREADY WAS, BECAUSE FLEXIBLE CONSTANTS ARE BY NO MEANS MORE  
FLEXIBLE THAN OTHER CONSTANTS.

NOTE THAT 'FLEX' CAN STILL APPEAR IN ACTUAL DECLARERS, AND THAT  
THE DECLARATION OF VARIABLES REMAINS THE ONLY PROPER WAY TO DECLARE  
A FLEXIBLE MULTIPLE.

#### C. UNITED MODES.

'VOID' IS NOT ALLOWED AS A CONSTITUENT OF UNITED MODES.  
THE 'VOID' DENOTATION 'EMPTY' HOWEVER WILL BE RECOGNIZED AS  
SUCH; ITS USE AS INDICANT WILL BE PRECLUDED IN ORDER TO DIS-  
ALLOW PROGRAMS THAT WOULD NOT BE ACCEPTABLE TO A COMPILER  
WITHOUT THIS RESTRICTION.

#### D. SCOPES.

A SCOPE IS ALWAYS SOME RANGE.  
ALL GENERATION YIELD VALUES OF GLOBAL SCOPE.  
NO SCOPE CHECK IS IMPLEMENTED ON ASSIGNMENT OR THE YIELD OF A  
RANGE OR ROUTINE. ROUTINES ARE SCOPE-CHECKED ONLY WHEN THEY ARE CALLED.

#### E. SUB SYMBOL AND BUS SYMBOL.

THE USE OF "STYLE ONE SUB SYMBOL" ("(") AND "STYLE ONE BUS SYMBOL" (")")  
IS NOT ALLOWED.

#### F. TRANSPUT.

WHICH IT ESSENTIALLY ALREADY WAS, BECAUSE FLEXIBLE CONSTANTS ARE BY NO MEANS MORE FLEXIBLE THAN OTHER CONSTANTS.

NOTE THAT 'FLEX' CAN STILL APPEAR IN ACTUAL DECLARERS, AND THAT THE DECLARATION OF VARIABLES REMAINS THE ONLY PROPER WAY TO DECLARE A FLEXIBLE MULTIPLE.

#### C. UNITED MODES.

'VOID' IS NOT ALLOWED AS A CONSTITUENT OF UNITED MODES. THE 'VOID' DENOTATION 'EMPTY' HOWEVER WILL BE RECOGNIZED AS SUCH; ITS USE AS INDICANT WILL BE PRECLUDED IN ORDER TO DISALLOW PROGRAMS THAT WOULD NOT BE ACCEPTABLE TO A COMPILER WITHOUT THIS RESTRICTION.

#### D. SCOPES.

A SCOPE IS ALWAYS SOME RANGE. ALL GENERATION YIELD VALUES OF GLOBAL SCOPE. NO SCOPE CHECK IS IMPLEMENTED ON ASSIGNMENT OR THE YIELD OF A RANGE OR ROUTINE. ROUTINES ARE SCOPE-CHECKED ONLY WHEN THEY ARE CALLED.

#### E. SUB SYMBOL AND BUS SYMBOL.

THE USE OF "STYLE ONE SUB SYMBOL" ("(") AND "STYLE ONE BUS SYMBOL" (")") IS NOT ALLOWED.

#### F. TRANSPUT.

1. BINARY AND NON-BINARY TRANSPUT CANNOT BE USED WITH THE SAME 'BOOK'.
2. "PUT" ON A "SEQUENTIAL FILE", SETS THE "LOGICAL END OF FILE" PAST THE LAST CHARACTER. THIS IMPLIES THAT ONLY IN THE LAST "LINE" OF A "SEQUENTIAL FILE" CHANGES CAN BE MADE.
3. IN THE CASE THAT "C OF CPOS" EQUALS 1, A "BACKSPACE" IS EQUIVALENT TO 'SKIP'.
4. THE SAME PARAMETERS APPLY TO BINARY TRANSPUT AND UNFORMATTED TRANSPUT.
5. THE "IDF" OF A FILE MUST BE A FOR SCOPE 3.4 ACCEPTABLE LOGICAL FILE NAME. EMPTY IDF'S ARE NOT ALLOWED.

THE COERCIONS TO MODE '^INTYPE' AND '^OUTTYPE' ARE DONE IN THE FOLLOWING WAY:

- 1 'MODE' '^IN' = 'CO' A. PLAIN  
B. ROW OF '^IN'  
C. STRUCT CONTAINING ONLY '^IN'  
'CO'

- 2 'MODE' '^INP' = 'CO' A. 'REF' '^IN'



## E. SUB SYMBOL AND BUS SYMBOL.

THE USE OF "STYLE ONE SUB SYMBOL" ("(") AND "STYLE ONE BUS SYMBOL" (")") IS NOT ALLOWED.

## F. TRANSPUT.

1. BINARY AND NON-BINARY TRANSPUT CANNOT BE USED WITH THE SAME 'BOOK'.
2. "PUT" ON A "SEQUENTIAL FILE", SETS THE "LOGICAL END OF FILE" PAST THE LAST CHARACTER.  
THIS IMPLIES THAT ONLY IN THE LAST "LINE" OF A "SEQUENTIAL FILE" CHANGES CAN BE MADE.
3. IN THE CASE THAT "C OF CPOS" EQUALS 1, A "BACKSPACE" IS EQUIVALENT TO 'SKIP'.
4. THE SAME PARAMETERS APPLY TO BINARY TRANSPUT AND UNFORMATTED TRANSPUT.
5. THE "IDF" OF A FILE MUST BE A FOR SCOPE 3.4 ACCEPTABLE LOGICAL FILE NAME. EMPTY IDF'S ARE NOT ALLOWED.

THE COERCIONS TO MODE '^INTYPE' AND '^OUTTYPE' ARE DONE IN THE FOLLOWING WAY:

- 1 'MODE' '^IN' = 'CO' A. PLAIN  
B. ROW OF '^IN'  
C. STRUCT CONTAINING ONLY '^IN'  
'CO'
- 2 'MODE' '^INP' = 'CO' A. 'REF' '^IN'  
B. COLLATERAL CLAUSE WITH ONLY '^INP'  
C. BALANCE WITH ONLY '^INP'  
D. UNION CONTAINING ONLY 'REF' '^IN'  
'CO'
- 3 'MODE' '^INTYPE' = 'CO' A STRUCT WITH FIELDS OF MODE '^INP' AND 'PROC'('REF' 'FILE') 'VOID' AND 'FORMAT'  
'CO'
- 4 'MODE' '^OUT' = 'CO' A. PLAIN  
B. ROW OF '^OUT'  
C. STRUCT OF '^OUT'  
D. UNION WITH ONLY '^OUT'  
'CO'
- 5 'MODE' '^OUTP' = 'CO' A. '^OUTP'  
B. COLLATERAL WITH ONLY '^OUTP'  
C. BALANCE WITH ONLY '^OUTP'  
'CO'

C. BALANCE WITH ONLY '^INP'  
D. UNION CONTAINING ONLY 'REF' '^IN'

'CO'

3 'MODE' '^INTYPE' = 'CO' A STRUCT WITH FIELDS OF MODE  
'^INF' AND  
'PROC'('REF','FILE') 'VOID' AND  
'FORMAT'

'CO'

4 'MODE' '^OUT' = 'CO' A. PLAIN  
B. ROW OF '^OUT'  
C. STRUCT OF '^OUT'  
D. UNION WITH ONLY '^OUT'

'CO'

5 'MODE' '^OUTP' = 'CO' A. '^OUTP'  
B. COLLATERAL WITH ONLY '^OUTP'  
C. BALANCE WITH ONLY '^OUTP'

'CO'

6 'MODE' '^OUTTYPE' = 'CO' A STRUCT WITH FIELDS OF MODE  
'^OUTP' AND  
'PROC'('REF','FILE') 'VOID' AND  
'FORMAT'

'CO'

#### SPECIAL CASES

- 1 JUMPS ARE ALLOWED.
- 2 'NIL' (IN SOURCE) IS TREATED AS OF MODE 'REF' '^INT'.
- 3 'SKIP' (IN SOURCE) IS TREATED AS OF MODE 'INT' (OUTPUT) OR  
'REF' '^INT' (INPUT)
- 4 VACUUM (IN SOURCE) IS TREATED AS OF MODE [] '^INT'.

#### G. FORMATS.

SEE APPENDIX.

H. NO FRAGMENTS ARE ALLOWED BETWEEN THE ALPHANUMERICAL CHARACTERS OF A TAG,  
BOLD TAG, THE NON-A.N. CHARACTERS COMPOSING AN OPERATOR OR  
THE CHARACTERS FORMING A DENOTATION.

#### I. RESTRICTIONS FOR THE PRE-RELEASE-VERSION.

FOR THE PRE-RELEASE-VERSION OF THE CYBER 70 ALGOL 68 COMPILER THE FOLLOWING  
RESTRICTIONS EXIST:

- FORMATTED TRANSPUT NOT IMPLEMENTED
- USER DEFINED LINE-MENDED PAGE-MENDED ETC. NOT IMPLEMENTED
- PARALLEL-CLAUSES NOT IMPLEMENTED
- 'LONG' AND 'SHORT' NOT IMPLEMENTED
- SCOPE-CHECK NOT IMPLEMENTED
- VERY LIMITED ERROR-RECOVERY

## I. RESTRICTIONS FOR THE PRE-RELEASE-VERSION.

FOR THE PRE-RELEASE-VERSION OF THE CYBER 70 ALGOL 68 COMPILER THE FOLLOWING RESTRICTIONS EXIST:

- FORMATTED TRANSPUT NOT IMPLEMENTED
- USER DEFINED LINE-ENDED PAGE-ENDED ETC. NOT IMPLEMENTED
- PARALLEL-CLAUSES NOT IMPLEMENTED
- 'LONG' AND 'SHORT' NOT IMPLEMENTED
- SCOPE-CHECK NOT IMPLEMENTED
- VERY LIMITED ERROR-RECOVERY

## II IMPLEMENTATIONS OF "UNDEFINED"

THE VALUE UNDEFINED IS USED OCCASIONALLY IN THE RR.

IN THE CYBER \* ALGOL 68 COMPILER THE FOLLOWING IMPLEMENTATIONS ARE CHOSEN:

1. AN ARRAY-BOUND ERROR IN SUBSCRIPTING, SLICING, ASSIGNMENT AND ROWING OF MULTIPLES CAUSES AN ERROR MESSAGE TO BE PRINTED.

2. IN GENERAL, AN ASSIGNMENT TO 'NIL' OR A DEREFERENCING OF 'NIL' CAUSES A "MODE-1 ERROR".

IF THE 'NIL' STANDS FOR A DESCRIPTOR OR A STRUCTURE WITH A FIRST FIELD OF A DESCRIPTOR, AN ERROR MESSAGE IS GIVEN.

3. ALL VALUES THAT ARE NOT INITIALIZED HAVE "SKIP".

4. - ASSIGNMENT TO 'SKIP' OR DEREFERENCE OF 'SKIP' SEE 2.2 (NIL)

- A 'REAL' OF VALUE 'SKIP' IS "INDEFINITE".
- A 'BOOL', 'CHAR', 'INT', 'BITS' OR 'BYTES' OR STRUCTURES WITH VALUE 'SKIP' USED IN FORMULA'S GIVES UNPREDICTABLE RESULTS
- A 'SKIP' AS 'INT' IN AN INTEGRAL CASE CLAUSE GIVES CONTROL TO THE OUT-CLAUSE
- A 'SKIP' AS A DESCRIPTOR ON THE RIGHT HAND SIDE OF AN ASSIGNMENT OR FOR SUBSCRIPTING, SLICING AND ROWING CAUSES AN ERROR MESSAGE.
- A 'SKIP' AS A 'UNION' IN A CONFORMITY CASE CLAUSE CAUSES A MODE-1 ERROR
- A 'SKIP' AS A DESCRIPTOR ON THE RIGHT-HAND-SIDE OF AN ASSIGNMENT OR FOR SUBSCRIPTING, SLICING AND ROWING CAUSES AN ERROR MESSAGE

5. THE USE OF 'REAL' VALUES GREATER THAN MAXREAL CAUSES A MODE-2 ERROR.

6. THE USE OF INDEFINITE 'REAL' VALUES, 'SKIP', OR DIVISION BY ZERO CAUSES A MODE-4 ERROR.

MODE-ERRORS ARE REPRIEVED AND A MESSAGE IS PRINTED INDICATING THE CAUSE OF THE ERROR WHEN POSSIBLE.

7. A MODE ERROR CAUSES AN ERROR MESSAGE AFTER JOB REPRIEVAL.

- 23 THE USE OF 'REAL' VALUES GREATER THAN MAXREAL CAUSES A MODE-2 ERROR.  
25 6. THE USE OF INDEFINITE 'REAL' VALUES, 'SKIP', OR DIVISION BY ZERO CAUSES  
A MODE-4 ERROR.  
27 MODE-ERRORS ARE REPRIEVED AND A MESSAGE IS PRINTED INDICATING THE CAUSE  
OF THE ERROR WHEN POSSIBLE.  
29 7. A MODE ERROR CAUSES AN ERROR MESSAGE AFTER JOB REPRIEVAL.

### III INTERNAL PRESENTATION

ALL VALUES ARE REPRESENTED BY:

1. PRIMITIVE MODES ARE 1 WORD EACH.  
'INT': THE ABSOLUTE VALUE MUST BE  $< 2^{48}$   
'REAL': THE NORMAL CYBER 70 FLOATING POINT 60 BIT WORD.  
'CHAR': 48 0-BITS FOLLOWED BY A 12-BIT INTEGER  
'BOOL': 60 1-BITS FOR 'TRUE', 60 0-BITS FOR 'FALSE'  
'BITS': 12 0-BITS FOLLOWED BY 48 1-BIT 'BOOLS'  
'BYTES': 12 0-BITS FOLLOWED BY 4 12-BIT 'CHAR'S'
2. 'LONG' PRIMITIVE MODES ARE 2 WORDS EACH WITH THE SAME FORMATS AS ABOVE.
3. STRUCTURES ARE REPRESENTED BY THE COLLECTION OF THE FIELDS.
4. REFERENCES ARE REPRESENTED BY 1 WORD WITH THE FOLLOWING FORMAT.  
0 - 11 SPECIAL BIT PATTERN: 4000B  
12 - 23 SIZE OF THE OBJECT - 1  
24 1-BIT IF THE OBJECT CONTAINS A REFERENCE OR A DESCRIPTOR  
25 - 41 RESERVED  
42 - 59 ADDRESS OF THE FIRST WORD OF THE OBJECT
5. MULTIPLES ARE REPRESENTED BY A DESCRIPTOR OF  $2 + 2 * \text{"DIMENSION"}$  WORDS  
THERE ARE 2 GENERAL DESCRIPTION WORDS WITH THE FOLLOWING FORMAT:

#### WORD 1:

- 0 - 11 SPECIAL BIT PATTERN: 3777 B
- 12 - 23 SIZE OF THE ELEMENT - 1  
24 1-BIT IF THE ELEMENT CONTAINS A REFERENCE OR A DESCRIPTOR
- 25 - 35 THE NUMBER OF DIMENSIONS  
36 1-BIT IF THE MULTIPLE IS FLEXIBLE
- 37 - 41 RESERVED
- 42 - 59 ADDRESS OF THE FIRST WORD OF THE FIRST ELEMENT

#### WORD 2:

- 0 - 11 AN INTEGER 0 LE COIM LE NUMBER OF DIMENSIONS, TO SIGNAL  
FOR HOW MANY SUBSCRIPTS THE ELEMENTS HAVE CONSECUTIVE  
ADDRESSES.
- 12 - 59 (ADDRESS OF LAST ELEMENT) - (ADDRESS OF THE FIRST ELEMENT)



WORD 2:

- 0 - 11 AN INTEGER 0 LE COIM LE NUMBER OF DIMENSIONS, TO SIGNAL FOR HOW MANY SUBSCRIPTS THE ELEMENTS HAVE CONSECUTIVE ADDRESSES.
- 12- 59 (ADDRESS OF LAST ELEMENT) - (ADDRESS OF THE FIRST ELEMENT + 1).

FOR EACH DIMENSION THERE ARE 2 WORDS WITH THE THE FOLLOING FORMAT:

WORD 1:

- 0 - 59 LOWER BOUND

WORD 2:

IN CASE THE MULTIPLE IS NOT EMPTY

- 0 - 35 UPPERBOUND - LOWER BOUND + 1
- 36- 59 THE DISTANCE BETWEEN TWO ELEMENTS THAT HAVE THE SAME SUBSCRIPTS BUT THIS ONE.

WORD 2 IN CASE THE MULTIPLE IS EMPTY:

- 0 - 59 AN INDEFINITE FLOATING POINT NUMBER WITH MANTISA 'EQ' UPPERBOUND.

6. 'PROC' IS REPRESENTED BY 1 WORD WITH THE FOLLOWING FORMAT:

- 0 - 5 6 0-BITS
- 6 - 23 STATIC LINK, I.E. THE ADDRESS REQUIRED TO ADDRESS THE GLOBAL VARIABLES.
- 24 - 41 RESERVED.
- 42 - 59 ADDRESS OF THE FIRST EXECUTABLE INSTRUCTION OF THE PROCEDURE.

7. 'UNIONS' ARE REPRESENTED BY A UNION WORD FOLLOWED BY THE VALUE OF THE ACTUAL SIZE, FOLLOWED BY ENOUGH PADDED WORDS THAT THE LARGEST CHOICE OF THE UNION FITS.

THE UNION WORD HAS THE FOLLOWING FORMAT:

- 0 - 23 SIZE OF THE LARGEST CHOICE - 1.
- 24 - 41 RESERVED.
- 42 - 59 UNION NUMBER, I.E. A DESCRIPTIVE INTEGER ABOUT THE MODE OF THE ACTUAL CONTENTS OF THE UNION.

8. 'SKIP' AND 'VACUUM' ARE REPRESENTED BY 60007777000000400000.

9. NIL IS REPRESENTED BY 60007777000000600000.

0 - 5 6 0-BITS

6 - 23 STATIC LINK, I.E. THE ADDRESS REQUIRED TO ADDRESS THE GLOBAL VARIABLES.

24 - 41 RESERVED.

42 - 59 ADDRESS OF THE FIRST EXECUTABLE INSTRUCTION OF THE PROCEDURE.

7. 'UNIONS' ARE REPRESENTED BY A UNION WORD FOLLOWED BY THE VALUE OF THE ACTUAL SIZE, FOLLOWED BY ENOUGH PADDED WORDS THAT THE LARGEST CHOICE OF THE UNION FITS.

THE UNION WORD HAS THE FOLLOWING FORMAT:

0 - 23 SIZE OF THE LARGEST CHOICE - 1.

24 - 41 RESERVED.

42 - 59 UNION NUMBER, I.E. A DESCRIPTIVE INTEGER ABOUT THE MODE OF THE ACTUAL CONTENTS OF THE UNION.

8. 'SKIP' AND 'VACUUM' ARE REPRESENTED BY 60007777000000400000.

9. NIL IS REPRESENTED BY 60007777000000600000.

#### IV ENVIRONMENT INQUIRIES IN STANDARD PRELUDE.

THE STANDARD PRELUDE OF THE COMPILER CONTAINS THE FOLLOWING INQUIRIES:

'INT' INT LENGTHS = 1,  
'INT' INT SHORTS = 1,  
'INT' MAX INT =  $2^{*48} - 1$ ,  
'INT' REAL LENGTHS = 1,  
'INT' REAL SHORTS = 1,  
'REAL' MAX REAL =  $2^{*1022} * (2^{*48} - 1)$ ,  
'REAL' SMALL REAL =  $2^{*(-47)}$ ,  
'INT' BITSLENGTHS = 1,  
'INT' BITSHORTS = 1,  
'INT' BITS WIDTH = 48,  
'INT' BYTES LENGTH = 1,  
'INT' BYTES SHORTS = 1,  
'INT' BYTES WIDTH = 4,  
'INT' MAXAB CHAR = 4095,  
'CHAR' NUL CHARACTER = 0,  
'CHAR' FLIP = 'T',  
'CHAR' FLOP = 'F',  
'CHAR' ERRORCHAR = '\*',  
'CHAR' BLANK = ' '.

IN THIS PRELUDE IS DECLARED:

'MODE' 'LONGBYTES' = 'STRUCT' {'BYTES' B1, B2};

IN THIS PRELUDE IS DECLARED:

```
'PROG' ('INT'M) 'INT' MEMORY = 'CO' IF THE PARAMETER IS LARGER THAN
THE CURRENT FIELD LENGTH AND NOT LAR-
GER THAN THE MAXIMUM FIELDLENGTH.
THEN INCREASE THE FIELDLENGTH TO THE
SIZE OF THE PARAMETER;
```

'INT' GARBAGE = 'CO' TOTAL NUMBER OF WORDS, FREED BY ALL GARBAGE  
COLLECTIONS 'CO',  
'REAL' COLLECT SECONDS = 'CO' TOTAL TIME IN CP SECONDS SPENT IN GARBAGE  
COLLECTING 'CO',

AND THE FOLLOWING PROCEDURES :

'PROC' ('INT' M) 'INT' MEMORY = 'CO' IF THE PARAMETER IS LARGER THAN  
THE CURRENT FIELD LENGTH AND NOT LAR-  
GER THAN THE MAXIMUM FIELDLENGTH,  
THEN INCREASE THE FIELDLENGTH TO THE  
SIZE OF THE PARAMETER;  
THE RESULT SIZE IS THE SIZE OF THE FIELDLENGTH 'CO',

'PROC' 'VOID' COLLECT GARBAGE = 'CO' DO A GARBAGE COLLECTION 'CO';

#### V. STACK/HEAP ORGANIZATION AND PARAMETER MECHANISM.

FOR EACH ACTIVE PROCEDURE THE STACK CONTAINS:

- RETURN-INFORMATION ( 1 WORD)
- PROCEDURE-HEADER ( 1 WORD)
- PARAMETERS
- STATIC WORKING STACK

FOR EACH ACTIVE RANGE THE STACK CONTAINS :

- RANGE HEADER ( 1 WORD)
- STATIC IDENTIFIER STACK
- STATIC WORKING STACK

THE STATIC PART OF EACH IDENTIFIER IS IN THE DYNAMIC PART IN  
THE HEAP. THIS IMPLIES THAT THE STATIC IDENTIFIER STACK, AFTER  
INITIALIZATION BY THE DECLARATION, IS NOT CHANGED ANYMORE.  
THE ADDRESSES OF THE IDENTIFIERS ARE CALCULATED USING THE OFFSET FROM THE PROCEDURE HEADER.

#### VI. CONTROL CARD FOR COMPILATION.

THE FOLLOWING CONTROL CARD INVOKES THE COMPILER:  
A68(P1,P2, ... PK) OR A68,P1,P2, ...,PK.

ALL PARAMETERS P1 THRU PK ARE NAME PARAMETERS AND NONE IS MANDATORY.

I : SOURCE INPUT.

OMITTED SOURCE INPUT FROM FILE INPUT.



THE HEAP. THIS IMPLIES THAT THE STATIC IDENTIFIER STACK, AT INITIALIZATION BY THE DECLARATION, IS NOT CHANGED ANYMORE. THE ADDRESSES OF THE IDENTIFIERS ARE CALCULATED USING THE OFFSET FROM THE PROCEDURE HEADER.

## VI. CONTROL CARD FOR COMPILATION.

THE FOLLOWING CONTROL CARD INVOKES THE COMPILER:  
A68(P1,P2, ... PK) OR A68,P1,P2, ...,PK.

ALL PARAMETERS P1 THRU PK ARE NAME PARAMETERS AND NONE IS MANDATORY.

### I : SOURCE INPUT.

OMITTED	SOURCE INPUT FROM FILE INPUT.
I	SOURCE INPUT FROM FILE COMPILE.
L=LFN	SOURCE INPUT FROM FILE LFN.
I=0	NO SOURCE INPUT.

### B : BINARY OR RELOCATABLE OUTPUT.

OMITTED	BINARY OUTPUT ON FILE LGO.
B	BINARY OUTPUT ON FILE LGO.
B=LFN	BINARY OUTPUT ON FILE LFN.
B=0	BINARY OUTPUT IS SUPPRESSED.

### L : SOURCE LISTING.

OMITTED	SOURCE LISTING AND DIAGNOSTICS ON FILE OUTPUT.
L	SOURCE LISTING AND DIAGNOSTICS ON FILE OUTPUT.
L=LFN	SOURCE LISTING AND DIAGNOSTICS ON FILE LFN.
L=0	DIAGNOSTICS ON FILE OUTPUT.

### O : OBJECT LISTING IN COMPASS-LIKE FORM.

OMITTED	NO OBJECT LISTING.
O	OBJECT LISTING ON FILE OUTPUT.
O=LFN	OBJECT LISTING ON FILE LFN.
O=0	NO OBJECT LISTING.

### LO : SOURCE AND OBJECT LISTING.

OMITTED	SOURCE LISTING AND DIAGNOSTICS ON FILE OUTPUT.
LO	SOURCE, DIAGNOSTICS AND OBJECT LISTING ON FILE OUTPUT.
LO=LFN	SOURCE, DIAGNOSTICS AND OBJECT LISTING ON FILE LFN.
LO=0	DIAGNOSTICS ON FILE OUTPUT.

### A : INLINE SUBSCRIPTING

OMITTED	SUBSCRIPTING IS DONE BY A BOUND-CHECKING RUNTIME ROUTINE.
A	SUBSCRIPTING IS DONE BY INLINE-CODE, NO BOUND-CHECKING IS PERFORMED.

# LO: SOURCE AND OBJECT LISTING.

OMITTED SOURCE LISTING AND DIAGNOSTICS ON FILE OUTPUT.  
 LO SOURCE, DIAGNOSTICS AND OBJECT LISTING ON FILE OUTPUT.  
 LO=LFN SOURCE, DIAGNOSTICS AND OBJECT LISTING ON FILE LFN.  
 LO=0 DIAGNOSTICS ON FILE OUTPUT.

## A: INLINE SUBSCRIPTING

OMITTED SUBSCRIPTING IS DONE BY A BOUND-CHECKING RUNTIME ROUTINE.  
 A SUBSCRIPTING IS DONE BY INLINE-CODE, NO BOUND-CHECKING IS PERFORMED.

## P: PRELUDE AND POSTLUDE.

PRELUDE AND POSTLUDE OVERLAY MODULE IS  
 TAKEN FROM THE RUNTIME LIBRARY.

OMITTED DEFAULT MODULE NAME

P IDEM

P=NAME PRELUDE NAME IS NAME FROM LIBRARY A68LIB.

P=LIB/NAME PRELUDE NAME IS NAME FROM LIBRARY LIB.

P=0 NO PRELUDE: THE DEFINITIONS OF PRIMITIVE MODES ARE  
 INTERNALLY GENERATED.

## N: COMPILER MODE.

OMITTED NORMAL MODE, THE SOURCE INPUT MUST CONTAIN A SINGLE  
 PARTICULAR PROGRAM.

N LIBRARY ADDITION MODE.

SOURCE INPUT CONTAINS

NAME: ( PRELUDE ADDITION 'PR' PROG 'PR' POSTLUDE ADDITION ) 'PR' ENDPOLAD 'PR'

NAME IS THE NAME OF THE PRELUDE AND MUST BE NOT LONGER THAN 6 CHARACTERS.

'PR' PROG 'PR' INDICATES THE PLACE WHERE A PARTICULAR PROGRAM SHOULD BE INSERTED.

THE BINARY OUTPUT MUST BE PLACED IN A USER LIBRARY (E.G. A68LIB)

AND CAN BE USED IN A SUBSEQUENT COMPILATION BY A CONTROL CARD P OPTION.

EXAMPLES:

PLOT: ( 'PROC' 'PLOT' = ... ; 'PR' PROG 'PR' 'SKIP' ) 'PR' ENDPOLAD 'PR'

FILE: 'BEGIN' 'FILE' NONSTAND; OPEN(NONSTAND,...); 'PR' PROG 'PR' STOP; CLOSE(NONSTAND 'END'

'PR' ENDPOLAD 'PR'

THE LATTER PRELUDE REDEFINES THE LABEL STOP.

N=0

SEPARATE COMPILATION MODE.

THE SOURCE INPUT CONTAINS:

PROC OR OPERATOR DEFINITIONS WITH RIGHTHANDSIDE

FORMAL MODE PACK, 'PR' XDEF NAME 'PR' UNIT

'PR' FEDX 'PR'.

E.G.

( 'OP' 'MAX' = ( 'REAL' A,B ) 'REAL' ; 'PR' XDEF MAXR 'PR'

'IF' A=B 'THEN' A 'ELSE' B 'FI' 'PR' FEDX 'PR' ; 'SKIP' )

THE PROC OR OPERATOR CAN BE USED IN A PRELUDE BY THE SAME DEFINITION WITH

'PR' XDEF NAME 'PR' UNIT 'PR' FEDX 'PR' REPLACED BY 'PR' XREF NAME 'PR'

FILE BEGIN FILE NONSTANDY OPENINGNONSTANDY  
 'PR' ENDPOLAD 'PR'  
 THE LATTER PRELUDE REDEFINES THE LABEL STOP.

N=0 SEPARATE COMPILATION MODE.

THE SOURCE INPUT CONTAINS:

PROC OR OPERATOR DEFINITIONS WITH RIGHTHANDSIDE  
 FORMAL MODE PACK, 'PR' XDEF NAME 'PR' UNIT  
 'PR' FEDX 'PR'.

E.G.

('OP' 'MAX' = ('REAL' A,B) 'REAL': 'PR' XDEF MAXR 'PR'  
 'IF' A=B 'THEN' A 'ELSE' B 'FI' 'PR' FEDX 'PR'; 'SKI')

THE PROC OR OPERATOR CAN BE USED IN A PRELUDE BY THE SAME DEFINITION WITH  
 'PR' XDEF NAME 'PR' UNIT 'PR' FEDX 'PR' REPLACED BY 'PR' XREF NAME 'PR'

C : NUMBER OF SIGNIFIANT CHARACTERS ON SOURCE INPUT LINE.  
 OMITTED 72 SIGNIFICANT CHARACTERS ON SOURCE INPUT LINE.  
 C 72 SIGNIFICANT CHARACTERS.  
 C=N N SIGNIFICANT CHARACTERS. ( N LE 130 )

F : BOLD CONVENTION.

OMITTED BOLD WORD REPRESENTED BY THEIR CHARACTERS ENCLOSED  
 IN STROP CHARACTERS '.

F BOLD WORD REPRESENTED BY THEIR CHARACTERS PRECEDED  
 BY STROP CHARACTER.

A BOLD WORD MUST BE FOLLOWED BY A CHARACTER THAT CANNOT  
 APPEAR IN A BOLD SEQUENCE.

### XIII HARDWARE REPRESENTATION.

#### A. INTERNAL REPRESENTATION OF SYMBOLS.

THE COMPILER INTERNALLY USES A 12-BIT CODE, ILO, THAT FULLFILLS  
 THE FOLLOWING REQUIREMENTS.

ILO CONTAINS ALL SINGLE PUNCH CHARACTERS MENTIONED IN THE REVISED  
 REPORT. ILO IS A SUPERSET OF ASCII.

ALL 8-BIT CHARACTERS NOT YET DEFINED ARE USED AS OTHER MONAD.

ALL OTHER CHARACTERS ARE STRING ITEM.

THIS RESULTS IN THE FOLLOWING TABLE FOR ILO:

ILO		ASCII SYMBOL
DECIMAL	OCTAL	
0-31	0-37	CONTROL LAYOUT SYMBOLS
32	40	BLANK SPACE SYMBOL, SPACE LAYOUT SYMBOL
33	41	! 'OTHER MONAD'

ILO      ASCII      SYMBOL  
 DECIMAL    OCTAL

0-31	0-37	CONTROL	LAYOUT SYMBOLS
32	40	BLANK	SPACE SYMBOL, SPACE LAYOUT SYMBOL
33	41	!	'OTHER MONAD'
34	42	"	QUOTE SYMBOL
35	43	#	STYLE TWO COMMENT SYMBOL
36	44	\$	FORMATTER SYMBOL
37	45	%	PERCENT SYMBOL
38	46	&	AMPERSAND SYMBOL
39	47	'	'OTHER MONAD'
40	50	(	OPEN SYMBOL, BRIEF BEGIN SYMBOL
41	51	)	CLOSE SYMBOL, BRIEF END SYMBOL
42	52	*	ASTERISK SYMBOL
43	53	+	PLUS SYMBOL
44	54	,	COMMA SYMBOL
45	55	-	MINUS SYMBOL
46	56	.	POINT SYMBOL
47	57	/	DIVIDED BY SYMBOL
48-57	60-71	0-9	DIGIT ZERO SYMBOL - DIGIT NINE SYMBOL
58	72	:	COLON SYMBOL, ROUTINE SYMBOL, UP TO SYMBOL, LABEL SYMBOL
59	73	;	GO ON SYMBOL
60	74	<	LESS THAN SYMBOL
61	75	=	EQUALS SYMBOL, IS DEFINED AS SYMBOL
62	76	>	GREATER THAN SYMBOL
63	77	?	'OTHER MONAD'
64	100	@	AT SYMBOL
65-90	101-132	A-Z	LETTER A SYMBOL - LETTER Z SYMBOL
69	105	E	TIMES TEN TO THE POWER SYMBOL
91	133	I	BRIEF SUB SYMBOL
92	134	\	TIMES TEN TO THE POWER SYMBOL
93	135	J	BRIEF BUS SYMBOL
94	136	^	'OTHER MONAD'
95	137	-	'OTHER MONAD'
96	140	-	'OTHER MONAD'
97-122	141-172		STYLE I LETTER A SYMBOL - STYLE I LETTER Z SYMBOL
101	145		TIMES TEN TO THE POWER SYMBOL
123	173		'OTHER MONAD'
124	174		STICK SYMBOL
125	175		'OTHER MONAD'
126	176		SKIP SYMBOL, TILDE SYMBOL
127	177	DEL	LAY OUT SYMBOL
128	200		OF SYMBOL
129	201		TIMES TEN TO THE POWER SYMBOL
130	202		DIFFERS FROM SYMBOL
131	203		UP SYMBOL
132	204		DOWN SYMBOL



126	176	SKIP SYMBOL, TILDE SYMBOL
127	177	DEL
128	200	LAY OUT SYMBOL
129	201	OF SYMBOL
130	202	TIMES TEN TO THE POWER SYMBOL
131	203	DIFFERS FROM SYMBOL
132	204	UP SYMBOL
133	205	DOWN SYMBOL
134	206	FLOOR SYMBOL
135	207	CEILING SYMBOL
136	210	AND SYMBOL
137	211	OR SYMBOL
138	212	TIMES SYMBOL
139	213	OVER SYMBOL
140	214	PLUS I TIMES SYMBOL
141	215	NIL SYMBOL
142	216	BRIEF COMMENT SYMBOL
143	217	WINDOW SYMBOL
144	220	IS AT MOST SYMBOL
145	221	IS AT LEAST SYMBOL
146-175	222-257	NOT SYMBOL
176-185	260-271	'OTHER MONAD'
186-192	272-300	BOLD DIGIT ZERO SYMBOL - BOLD DIGIT
193-218	301-332	NINE SYMBOL
219-224	333-340	'OTHER MONAD'
225-250	341-372	BOLD LETTER A SYMBOL - BOLD LETTER
251-255	373-377	Z SYMBOL
256-4095	400-7777	'OTHER MONAD'
		'OTHER STRING ITEM'

## B. EXTERNAL REPRESENTATION.

THE 63-CHARACTER ASCII SET DOES NOT CONTAIN ALL SINGLE PUNCH CHARACTERS IN THE REVISED REPORT.

CHARACTERS APPEARING IN BOTH ASCII AND THE RR ARE MAPPED ONTO EACH OTHER.

CHARACTERS APPEARING IN ASCII, BUT NOT IN THE RR ARE OTHER MONAD. CHARACTERS IN THE RR, NOT IN ASCII ARE REPRESENTED BY THE BOLD SEQUENCE APPEARING THE RR.

HOWEVER ! IS USED AS STICK SYMBOL.

SPACE- AND SPACELAYOUT SYMBOL ARE REPRESENTED BY THE ASCII SPACE SYMBOL.

ALL POSSIBLE ILO SYMBOLS CAN BE PRESENTED AT ANY PLACE BY THE "GENERAL IMAGE":  
'(N)' OR '(N1,N2,.....NK)' WHERE THE NI ARE THE ILO REPRESENTATIONS IN DECIMAL NOTATION.

THE 63-CHARACTER ASCII SET DOES NOT CONTAIN ALL SINGLE PUNCH CHARACTERS IN THE REVISED REPORT.

CHARACTERS APPEARING IN BOTH ASCII AND THE RR ARE MAPPED ONTO EACH OTHER. CHARACTERS APPEARING IN ASCII, BUT NOT IN THE RR ARE OTHER MONAD. CHARACTERS IN THE RR, NOT IN ASCII ARE REPRESENTED BY THE BOLD SEQUENCE APPEARING THE RR. HOWEVER ! IS USED AS STICK SYMBOL. SPACE- AND SPACELAYOUT SYMBOL ARE REPRESENTED BY THE ASCII SPACE SYMBOL.

ALL POSSIBLE ILO SYMBOLS CAN BE PRESENTED AT ANY PLACE BY THE "GENERAL IMAGE": '(N)' OR '(N1,N2,.....NK)' WHERE THE NI ARE THE ILO REPRESENTATIONS IN DECIMAL NOTATION.

AS SINGLE PUNCHABLE CHARACTERS WE THEN HAVE:

OCTAL	ASCII	SYMBOL
40	BLANK	SPACE SYMBOL, SPACE LAYOUT SYMBOL
41	!	STICK SYMBOL
42	"	QUOTE SYMBOL
43	#	STYLE TWO COMMENT SYMBOL
44	\$	FORMATTER SYMBOL
45	%	PERCENT SYMBOL, NOT IN 63-ASCII
46	&	AMPERSAND SYMBOL
47	'	BOLD GLYPH
50	(	OPEN SYMBOL, BRIEF BEGIN SYMBOL
51	)	CLOSE SYMBOL, BRIEF END SYMBOL
52	*	ASTERISK SYMBOL
53	+	PLUS SYMBOL
54	,	COMMA SYMBOL
55	-	MINUS SYMBOL
56	.	POINT SYMBOL
57	/	DIVIDED BY SYMBOL
60-61	0-9	DIGIT ZERO SYMBOL - DIGIT NINE SYMBOL
72	:	COLON SYMBOL, UP TO SYMBOL, LABEL SYMBOL, ROUTINE SYMBOL
73	;	GO ON SYMBOL
74	<	LESS THAN SYMBOL
75	=	EQUALS SYMBOL, IS DEFINED AS SYMBOL
76	>	GREATER THAN SYMBOL
77	?	'OTHER MONAD'
100	@	AT SYMBOL
101-132	A-Z	LETTER A SYMBOL - LETTER Z SYMBOL
133	[	BRIEF SUB SYMBOL
134	\	TIMES TEN TO THE POWER SYMBOL
135	]	BRIEF BUS SYMBOL
136	^	'OTHER MONAD'
137	_	'OTHER MONAD'

		SYMBOL
73	;	GO ON SYMBOL
74	<	LESS THAN SYMBOL
75	=	EQUALS SYMBOL, IS DEFINED AS SYMBOL
76	>	GREATER THAN SYMBOL
77	?	'OTHER MONAD'
100	@	AT SYMBOL
101-132	A-Z	LETTER A SYMBOL - LETTER Z SYMBOL
133	[	BRIEF SUB SYMBOL
134	\	TIMES TEN TO THE POWER SYMBOL
135	]	BRIEF BUS SYMBOL
136	^	'OTHER MONAD'
137	-	'OTHER MONAD'

## PRAGMATS.

### 1. SEPARATE COMPILATION.

A ROUTINE TEXT MAY BE SEPARATELY COMPILED BY ENCLOSING ITS UNIT IN THE PRAGMATS 'PR' XDEF NAME 'PR' AND 'PR' FEDX 'PR' AND USING THE N OPTION ON THE A68 CONTROL CARD. NAME IS THE ENTRY-POINT NAME THAT WILL BE GENERATED FOR THE ROUTINE.

THE SOURCE TEXT FOR SEPARATE COMPILATION CONSISTS OF ONE OR MORE IDENTITY OR OPERATOR DECLARATIONS WITH ON THE RIGHT HAND SIDE A ROUTINE TEXT,

OF WHICH THE UNIT IS SURROUNDED BY THE ABOVE PRAGMATS.

THE LAST DECLARATION IS FOLLOWED BY 'SKIP'.

AN IDENTIFIER OR OPERATOR OF MODE 'PROC' ( ) MAY BE DEFINED AS EXTERNAL BY A DECLARATION OF THE FORM:

'OP' OPERATOR = ... OR

'PROC' IDENTIFIER = FORMAL PARAMETER PACK

'PR' XREF NAME 'PR' 'SKIP'

WHERE NAME IS THE ENTRY-POINT FOR THE ROUTINE.

### 2. LIBRARY ADDITION MODE.

'PR' PROG 'PR' IS USED TO INDICATE THE PARTICULAR PROGRAM IN LIBRARY ADDITION MODE.

### 3. LIST CONTROL.

'PR' LIST 'PR' TURN SOURCE LISTING ON.

'PR' NOLIST 'PR' TURN SOURCE LISTING OFF.

'PR' EJECT 'PR' PAGE EJECT ON SOURCE LISTING.

### 4. SOURCE INPUT BOLD CONVENTION CONTROL.

'PR' FLAGGED 'PR' SWITCH TO FLAGGED MODE (AS UNDER F-OPTION).

'PR' LIST 'PR' TURN SOURCE LISTING ON.  
'PR' NOLIST 'PR' TURN SOURCE LISTING OFF.  
'PR' EJECT 'PR' PAGE EJECT ON SOURCE LISTING.

#### 4.SOURCE INPUT BOLD CONVENTION CONTROL.

'PR' FLAGGED 'PR' SWITCH TO FLAGGED MODE (AS UNDER F-OPTION).  
'PR' STROPPED 'PR' SWITCH TO STROPPED MODE (AS WITH F-OPTION OMITTED).

1\*\*\*\*\* A68-BULLETIN 100 \*\*\*\*\* DATE 03-10-74 TIME 09/44/17

DE HUIDIGE VERSIE IS DE OFFICIELE PRERELEASE VERSIE.

DE FORMATLESS TRANSPUT WERKT REDELIJK

CM67700 IS NODIG (ZEER VOORLOPIG) VOOR COMPILATIE

WIM GRIFFIOEN IS ERG GEINTERESSEERD IN UW PROBLEMEN!!!

GELIEVE ZO VEEL MOGELIJK DEZE VERSIE UIT TE TESTEN

MET ONBEGRIJPELIJKHEDEN RICHT MEN ZICH TOT W.GRIFFIOEN

TIJDELIJKE RESTRICTIES VOOR DE CDC A68-COMPILER.

ALLE BEKENDE RESTRICTIES.

\*\*\*\* 1 \*\*\*\* 09/05/74

PRINT ACCEPTEERT NIET DE VOLLEDIGE OUTTYPE ALS PARAMETER  
IEDERE PROC(REF FILE)VCID WORDT GEINTERPRETEERD ALS  
NEWLINE. ZIE OOK NO. 9

\*\*\*\* 2 \*\*\*\* 09/05/74

GECORRIGEERD 21/05/74.

\*\*\*\* 3 \*\*\*\* 09/05/74

GECORRIGEERD 21/05/74.

\*\*\*\* 4 \*\*\*\* 09/05/74

GECORRIGEERD . 10/05/74



GECORRIGEERD 21/05/74.

\*\*\*\* 4 \*\*\*\* 09/05/74

GECORRIGEERD . 10/05/74

\*\*\*\* 5 \*\*\*\* 09/05/74

GECORRIGEERD . 15/05/74

\*\*\*\* 6 \*\*\*\* 09/05/74

GECORRIGEERD 31/05/74

\*\*\*\* 7 \*\*\*\* 10/05/74

GECORRIGEERD 21/05/74

\*\*\*\* 8 \*\*\*\* 21/05/74

GECORRIGEERD 31/05/74

\*\*\*\* 9 \*\*\*\* 31/05/74

CHARACTERSET:

ZIE PUBLICATIE: HARDWARE REPRESENTATION EN ILO REPR.

CONTROLKAARTEN:

ATTACH, A68, ID=UACCU.

ATTACH, ZZZZIL0, ID=UACCU.

A68(OPTIONS) (?) ?

ZIE PUBLICATIE VOOR OPTIONS EN RESTRICTIES

ATTACH, A68LIB, ID=UACCU.

REDUCE OFF.

LGO.

EXIT.

CATALOG, LGO, ID=CDC.

COMPILER MODES:

1. NORMAL MODE: N OPTIEN NIET GESPECIFIEERD

2. LIBRARY ADDITION MODE: A. N

B. N=LOGICAL FILE NAME

OPMERKING: N IS EQUIVALENT MET N=ZZZZLIB

3. SEPERATE COMPILATION: N=0

OPMERKING: BIJ ONTBREKEN VAN "XDEF'S" IN DE SOURCE JOEKT

ATTACH,A68,ID=UACCU.  
ATTACH,ZZZZILO,ID=UACCU.  
A68(OPTIONS)()?)  
ATTACH,A68LIB,ID=UACCU.  
REDUCE OFF.  
LGO.  
EXIT.  
CATALOG,LGO,ID=CDC.

ZIE PUBLICATIE VOOR OPTIONS EN RESTRICTIES

#### COMPILER MODES:

1. NORMAL MODE: N OPTION NIET GESPECIFIEERD

2. LIBRARY ADDITION MODE: A. N

B. N=LOGICAL FILE NAME

OPMERKING: N IS EQUIVALENT MET N=ZZZZLIB

3. SEPERATE COMPILATION: N=0

OPMERKING: BIJ ONTBREKEN VAN "XDEF'S" IN DE SOURCE WERKT  
DE COMPILER ALS EEN TEXTTESTER.

CM EN T OP DE JOBKAART:

CM67700,T150.

#### A68 CONTROLCARD OPTIES (UITVOERIG):

A68(P1,P2,.....PN)()?)  
A68,P1,P2,.....PN.()?)

#### PARAMETERS PK:

1. SOURCE INPUT-PARAMETER : I (PARTICULAR PROGRAM)  
AFWEZIG : SOURCE INPUT VAN DE FILE INPUT.  
I=LFN : SOURCE INPUT VAN DE FILE LFN.  
I=0 : GEEN SOURCE INPUT.

2. AANTAL SIGNIFICANTE CHARACTERS OP EEN INPUTLIJN : C  
AFWEZIG : 72 (6 BITS) CHARACTERS.  
C : IDEM.  
C=N N < 130 : N CHARACTERS.

3. AANTAL LONGS: E  
AANTAL SHORTS: S  
ANDERS DAN E=0,S=0 MAG NIET WORDEN OPgegeVEN  
DE IMPLEMENTATIE ACCEPTEERT GEEN LONG EN SHORT

4. SOURCE LISTING: L  
AFWEZIG : SOURCE LISTING EN DIAGNOSTIEKEN VERSCHIJNEN OP  
FILE OUTPUT.  
L=LFN : OP FILE LFN.

1 IDEN.  
2 N CHARACTERS.

```

4. SOURCE LISTING: L
   AFWEZIG      : SOURCE LISTING EN DIAGNOSTIEKEN VERSCHIJNEN OP
                  FILE OUTPUT.
   L=LFN        : OP FILE LFN.
   L=0          : LIST OUTPUT ONDERDRUKT.

```

IN HET PROGRAMMA KAN DE STROP MODE WORDEN GEWIJZIGD DOOR:  
'PR STROPPED' PR  
EN 'PR FLAGGED' PR ,

COMPILATIE VAN MEER DAN EEN PROGRAMMA.

```
DE SOURCE INPUT FILE KAN NIET WORDEN GELEZEN ACHTER EEN  
END OF RECORD.  
MEERDERE PROGRAMMAS KUNNEN IN EEN KEER WORDEN GECOMPILEERD  
ALS ZE WORDEN GESCHEIDEN DOOR  
'PR' STOP 'PR'
```

**DMP'X.**

[illegible]

02644	12771	51400	00043	54710	12742	20600	01000	02651
02650	01000	02660	61000	46000	43014	51100	00426	11181
02654	61400	00073	01000	02660	51700	00427	04000	02631
02660	00000	00000	61000	46000	67442	43001	23040	11701
02664	36423	51100	03342	10611	52647	77776	61000	46001
02670	51200	00101	01000	02713	04000	04550	00000	00001
02674	54221	54321	54431	63640	21422	43560	15445	37521
02700	04000	02702	61000	46000	36554	03250	02671	46001
02704	04000	02671	61000	46000	51500	06607	74110	12111
02710	43352	76511	36435	11641	15523	36665	11232	74111
02714	20173	03010	02713	46000	03210	02705	22111	46001
02720	01300	00000	61000	46000	04000	02621	00000	00001
02724	51500	02730	10655	46000	51500	00066	03250	02721
02730	56510	03150	02720	46000	01300	00000	61000	46001
02734	51200	01104	43072	63520	03010	02762	36110	53411
02740	21701	03170	02735	73610	51600	01055	51100	01001
02744	73651	63420	61300	00003	67443	07400	02750	53211
02750	71100	00003	51000	00003	01000	03763	61000	46001
02754	51100	01055	51200	01104	66311	63213	53122	10711
02760	01000	05413	61000	46000	51100	01055	04000	02731
02764	01000	05121	61000	46000	43774	51700	01041	46001
02770	51300	01103	51400	01130	43566	66211	63540	66611
02774	04000	02772	61000	46000	04000	10322	00000	00001
03000	61200	00015	61400	00162	61700	00012	64510	66611
03004	03030	03015	07670	03002	04000	02775	61000	46001
03010	51600	00735	61700	00006	61600	00733	01000	04311
03014	04000	02561	61000	46000	51200	00711	03020	03021
03020	61600	00650	01000	04317	51100	06424	01000	02721

27/01/75 ACCU SCOPE 3.4.1 LV 373.6 13/01/75

13.17.03.SELC2H FROM \*\*

13.17.03.IP 00000192 WORDS - FILE INPUT , DC 00

13.17.04.SELC.

13.17.07.ACCOUNT,\*\*\*\*\*.

13.17.11.ATTACH,A68DJC,ID=UACCU,MR=1.

13.17.11.PFN IS

13.17.11.A68DJC

13.17.12.PF CYCLE NO. = 003

13.17.13.COPYSBF,A68DJC.

13.17.19.ATTACH,A68BULL,ID=UACCU,MR=1.

13.17.19.PFN IS

13.17.19.A68BULL

13.17.20.PF CYCLE NO. = 001

13.17.20.COPYSBF,A68BULL.

13.17.23.ATTACH,OLOPL,ACCULIB,ID=UACCU,CY=2,MR=1.

13.17.23.

13.17.44.UPDATE,Q,D,L=0.

13.17.46. 1 ERRORS IN UPDATE INPUT

13.17.51. 1 UPDATE ERRORS, JOB ABORTED.

13.17.53.OP 00004544 WORDS - FILE OUTPUT , DC 40

13.17.53.MS 7168 WORDS ( 7168 MAX USED)

13.17.53.CPA .293 SEC. .293 ADJ.

13.17.53.CPB .334 SEC. .334 ADJ.

13.17.53.IO .434 SEC. .434 ADJ.

13.17.53.CM 11.559 KWS. .704 ADJ.

13.17.53.SS 1.768

13.17.53.PP 14.001 SEC. DATE 27/01/75

13.17.53.MAX CM USED 20000B WORDS

13.17.53.EJ END OF JOB, \*\*